

# Visualization to compare karate motion captures

Dana K. Urribarri<sup>1</sup>, Martín L. Larrea<sup>1,2</sup>, Silvia M. Castro<sup>1,2</sup>, and Enrico Puppo<sup>3</sup>

<sup>1</sup> VyGLab (CIC-UNS)

<sup>2</sup> ICIC (UNS-CONICET)

Depto de Cs. e Ing. de la Computación (UNS)

San Andrés 800, 8000 Bahía Blanca, Buenos Aires, Argentina

{dku, mll, smc}@cs.uns.edu.ar

<sup>3</sup> DIBRIS

University of Genoa, Genoa, Italy

enrico.puppo@unige.it

**Abstract.** Multi-dimensional time series from motion capture (MoCap) provide a rich source of data for human motion analysis, yet they are difficult to process and compare. We address MoCap data related to Karate *katas*, containing predefined sequences of movements, executed independently by several subjects with different timing and speed. We propose a combination of signal processing and data visualization techniques to analyze the misalignment between data from different subjects. We present a web app that implements this proposal, providing a visual comparison of time series, based on Dynamic Time Warping.

**Keywords:** Dynamic Time Warping, Comparative Visualization, Data Visualization, Multidimensional Data, Motion Capture

## 1 Introduction

Motion capture (MoCap) data are becoming increasingly important to understand human movements. MoCap data analysis can be used, e.g., to find movements in a database [13], or to compare how different people move [15]. While some tasks are suited for automatic analysis, others may be better performed visually: Bernard et al. [3] emphasize the importance of visual comparison techniques and the challenge to design them. Even though a video rendering provides realistic visualization, comparing movements from different videos is a rather difficult and time-consuming task [14, 15]. Conversely, proper data visualization techniques can provide information about a whole MoCap sequence at a glance.

In this paper, we present a visualization tool based on Dynamic Time Warping, which supports visual comparison between two similar sequences of movements to analyze their time alignment. In order to address codified sequences characterized with neat movements, we take karate *katas* performed by various participants as a case study. Our visualization shows how *source* sequence of movements must be transformed to match a *reference* sequence. This allows, e.g., comparing the performance of karate students with respect to a sequence performed by the teacher, considered to be correct.

## 2 Previous work

### 2.1 MoCap visualization

Visualizing a whole MoCap is not an easy task. Li et al. [14] present a review of MoCaps visualization techniques, providing a classification into images summaries, interactive platforms, and animations. Image summaries rely mostly on keyframes [1, 9], i.e., frames that best represent motion in a time window. Malmstrom et al. [15] visualize and compare just a joint from two MoCaps and also include a skeletal view. Hu et al. [9] propose a subject-centered representation called *motion track*, a matrix that only represents angles between skeleton parts, disregarding translation and orientation information; this matrix supports both analyzing one motion sequence and comparing multiple motion sequences. Dimensionality reduction is another technique proposed to visualize time series and MoCaps [20]; however, it is proposed only as a complement for visualization tools, since it helps reveal meaningful information about data clusters but not on interpreting individual information or inspecting changes over time.

As Bernard et al. [3] state, the design of visual strategies to compare MoCaps is also full of challenges regarding data preprocessing, visualization design and interactions. Multiple visualizations focus on comparing data reduced to one 2D or 3D trajectory [4, 8, 9, 13, 22], comparison of patterns [2, 10] or overlapping of keyframes [10]. More recently, Sedmidubsky et al. [19] compare the movements using convolutional neural networks, so the 3D position of the joints are represented as an image. However, they do not intend this image to be part of a visualization system. Just a few visualizations permit to interactively synchronize sequences [22].

### 2.2 Comparison of 3D trajectories

In the signal processing field, Dynamic Time Warping (DTW) is a very well-known technique to compare two signals; for example, it is commonly used for speech recognition [18]. Bruderlin and Williams [6] stated that signal processing methods, including DTW [5] are suitable for motion data represented with segment lengths and joint angles.

Nowadays, several works use DTW to synchronize and compare movement data [16]. Krüger et al. [13] use DTW as part of a bigger system to find similarities between movements in a database comparing the 3D position of the joints. Vaughan and Gabrys [21] use DTW to analyze trajectory data ( $x$ ,  $y$ ,  $z$ , force, and pressure) from virtual reality haptic training simulator for epidural needle insertion and evaluate the skill level of trainees; DTW allows them to compare the similarity between two trajectories and to combine together two or more trajectories to produce an average trajectory. Jiang et al. [11] use DTW to evaluate surgical skills for robot-assisted minimally invasive surgery; they propose an additional indicator to include large deviations and velocity changes in different phases of the training tasks into analysis.

### 3 Data preparation

This work is based on the motion captures corresponding to the Bassai Dai kata of the karate dataset<sup>1</sup> presented in [12] and [17]. The Bassai Dai dataset consists of 20 MoCap sequences by 7 participants with different skills and age. Each MoCap sequence contains information about 25 markers placed on the participant's body (see figure 1). The participants are 2 adults with very high expertise (a Karate teacher and a participant to international Championships); 2 black-belt adults with high expertise; and 4 teenagers with medium expertise. Except for one of the black-belt adults, all are men.

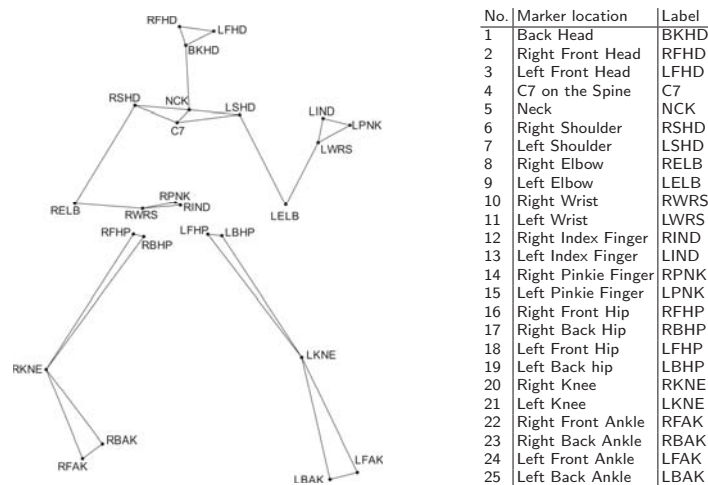


Fig. 1: Labels and positions of the 25 markers that form the MoCaps.

#### 3.1 Data completion

In the skeleton from figure 1, markers are mutually connected with edges that resemble human bones. MoCap sequences are affected by noise and missing data. To complete the missing values of markers, we used a simple strategy considering that the distance between two markers connected by a bone remains constant. For incomplete bones, the position of a missing marker is estimated interpolating the orientation of the bone.

#### 3.2 Data normalization

Data normalization is an inherent problem of sequence comparison. Several approaches exist in the literature. In this work we used an approach similar to the

<sup>1</sup> [http://www.infomus.org/karate/eyesweb\\_dataset\\_karate\\_eng.php](http://www.infomus.org/karate/eyesweb_dataset_karate_eng.php)

one presented by Sedmidubsky et al. [19]; they normalize the data orientation and the size of the skeleton transforming with rotation and translation the 3D position of the joints.

**Skeleton alignment.** To normalize all the sequences, the skeleton of every frame was transformed in a way that the joint NCK is fixed at point  $(0,0,0)$ , the segment formed by joints NCK–BKHD is aligned with axis  $z$  and the triangle formed by joints NCK–C7–BKHD is aligned with plane  $yz$ . Information about the roto-translation of the body with respect to the world reference system is lost; however, to compare the timing of each pose, this information resulted not important.

**Space normalization.** Finally, in the last step of data preparation, the whole skeleton-aligned MoCap is translated and scaled to fit into the box  $(0,0,0)–(1,1,1)$  while the joint NCK is fixed at point  $(0.5,0.5,0.5)$ . This step reduces distortions due to the different heights of participants.

**Data reduction.** The basic implementation of DTW needs that both sequences have the same number of frames. Since the captures were taken at a rate of 250Hz and produced sequences of about 25000 frames, every sequence was downsampled, with the Lanczos resampling [7], to have all the same number of frames and a rate of at least 25Hz. In this context, each frame represents a variable amount of time. However, this allows us to compare movements of the sequence and its proportional speeds, regardless of the actual speed at which they were performed.

## 4 Solution design

We propose a visualization tool that combines different techniques to show how misaligned (hastened or delayed) a *source* MoCap sequence is with respect to a *reference*. In this section  $A$  represents the reference MoCap and  $B$  represents the source MoCap.

### 4.1 Time alignment

We use DTW to align in time the two sequences. This technique provides not only a measure of dissimilarity, but also a warping function and, consequently, a misalignment function. DTW computes a mapping between the two sequences that is the optimal match between them. Time warping induced by this mapping provides a measure of dissimilarity between the two sequences. Furthermore, one of the sequences can be transformed by this mapping to obtain two time-aligned sequences.

**A dissimilarity measure.** The obtain the measure of dissimilarity between to sequences  $A$  and  $B$ , DTW computes the accumulated distance matrix  $M$  between them (see Algorithm 1); that is, it compares every frame of  $A$  with every frame of  $B$ . The accumulated distance  $M_{L,L}$  represents the dissimilarity between the two series and is a measure of the cost of warping one of the sequences to match the other. Euclidean distance is usually used to calculate the distance between frames, and it is the one used in this design; however, any suitable distance function can be used.

---

**Algorithm 1** Accumulated distance matrix  $M$  between sequences  $A$  and  $B$

---

**Input** Two sequences  $A$  and  $B$  with  $L$  frames each one.

**Output** The accumulated distance matrix  $M$  between the two sequences.

---

```

    ▷ Let  $X_n$  be the frame  $n$  of sequence  $X$ .
1:  $M_{1,1} \leftarrow \text{dist}(A_1, B_1)$            ▷ The distance between the initial frames
2: for all  $i \in [2, L]$  do
    ▷ Accumulated distance between sequence  $B$  and the first frame of sequence  $A$ .
3:    $M_{1,i} \leftarrow \text{dist}(A_1, B_i) + M_{1,i-1}$ 
    ▷ Accumulated distance between sequence  $A$  and the first frame of sequence  $B$ .
4:    $M_{i,1} \leftarrow \text{dist}(A_i, B_1) + M_{i-1,1}$ 
5: for all  $i, j \in [2, L]$  do
    ▷ Minimum accumulated distance between every other pair of frames
6:    $M_{i,j} \leftarrow \text{dist}(A_i, B_j) + \min\{M_{i-1,j-1}, M_{i-1,j}, M_{i,j-1}\}$ 

```

---

**Warping function.** The computation of the measure  $M_{L,L}$  implies finding the minimal accumulated-distance path from  $(1, 1)$  to  $(L, L)$ . This *warping path*  $P$  is a list of pairs  $(i, j)$  that defines a warping from sequence  $B$  into sequence  $A$ : each frame  $i$  in  $B$  is warped into a frame  $j$  in  $A$ . Figure 2 shows the accumulated distance matrix between two sequences and the warping path between them. Ideally, this warping is a function that maps every frame of  $B$  into some frame of  $A$ . However, in the warping path, there may exist several pairs warping the same frame  $i$  of  $B$  into different frames of  $A$ . Then, a function  $F : N \rightarrow N$  that warps sequence  $B$  into  $A$  is defined as:

$$F(x) = \lfloor \text{mean}_{(x,j) \in P} \{j\} \rfloor \quad (1)$$

## 4.2 The misalignment function

Given a frame  $n$  of sequence  $B$ , the function  $F(n)$  gives the time frame of  $A$  that corresponds to it. The misalignment function  $G(n)$  from Equation 2 indicates the distance in frames from  $n$  to  $F(n)$ :

$$G(n) = F(n) - n \quad (2)$$

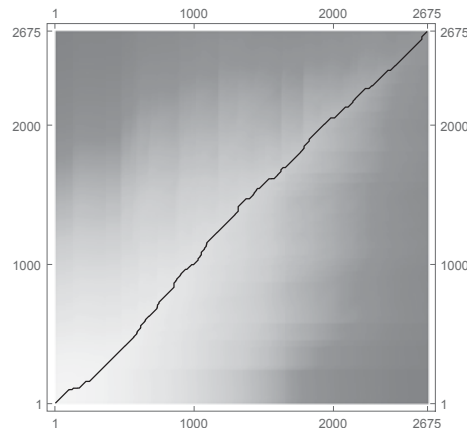


Fig. 2: Accumulated distance matrix and warping path between two sequences of 2675 frames. Graylevel represents distance, the darker the larger. The value of  $M_{2675,2675}$  provides a measure of dissimilarity between sequences. Columns represent the frames from sequence  $B$  and rows, the frames from sequence  $A$ .

If for two consecutive time frames  $n$  and  $n + 1$ ,  $G(n + 1)$  is greater than  $G(n)$  (positive slope), then  $B$  is hastened. Conversely, if  $G(n + 1)$  is smaller than  $G(n)$ , then  $B$  is delayed. The ideal scenario is when  $G(n + 1)$  and  $G(n)$  are equal at all frames, which means that  $B$  is perfectly synchronized with  $A$ .

### 4.3 Heat map visualization

In a heat map, the color of each pixel encodes information. In this case, the color encodes the position  $(x, y, z)$  of each marker: the red, green and blue channels encode the  $x$ ,  $y$  and  $z$  coordinates, respectively. Since after space normalization all the positions of the markers lie in the box  $(0,0,0)-(1,1,1)$ , then the representation as a color is straightforward. One dimension of the image represents the timeline; this means that adjacent pixels in this dimension represent consecutive positions of the same marker. The other dimension represents the markers. In this dimension, adjacency of pixels is not relevant. Figure 3a shows the representation with colors of the pose of Figure 1, and Figure 3b shows a whole MoCap visualized as only one image.

## 5 The proposed visualization

In order to show the amount of synchronization between two MoCaps, we designed a web visualization that implements the previous design. Figure 4 provides a view of our visualization tool. The left panel shows a table with all available MoCaps to compare; once that the reference and source MoCaps are selected, the *Visualize* button shows the visualization in the right panel.

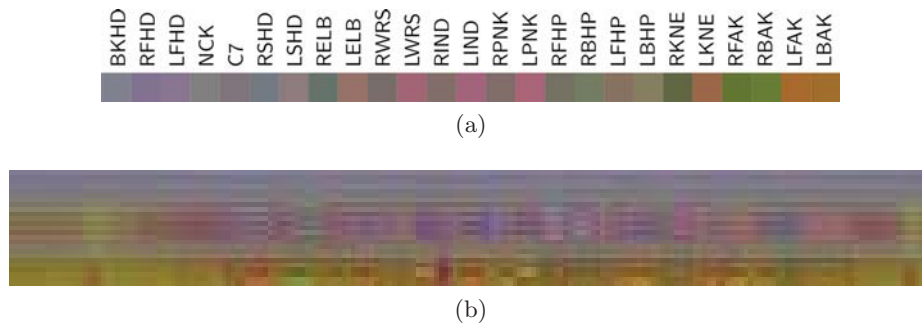


Fig. 3: Heat map visualization of a MoCap. **(a)** Color representation of the pose of Figure 1. The color of each pixel is given by the position of the marker inside the  $(0,0,0)$ – $(1,1,1)$  box. **(b)** Color representation of a whole MoCap. The row at the top corresponds to marker BKHD and the one at the bottom, to marker LBAK. Times increases from left to right: the first column on the left corresponds to the first frame of the MoCap, and the last column on the right, to the last frame of the MoCap.

The presented visualizations compare the MoCaps corresponding to the performances of two participants, or two different performances of the same participant; the graphs show differences between the source and the reference:

1. How delayed or hastened each frame of the source is with respect to the reference. A color scale encodes this information: red means delayed, blue means hastened and yellow means on time. The  $y$  coordinate of this graph represents the absolute difference between the poses of time-aligned frames. This difference measures the Euclidean distance between markers, considering the collection of  $N$  markers as a unique point in a  $3N$ -dimensional space. This information gives an idea of how similar the poses are.
2. The misalignment function  $G(n)$  between source and reference (Equation 2). The  $x$ -axis corresponds to the timeline of the source MoCap. Positive slopes indicate that the source performance is hastened and leaped forward in time with respect to the reference and negative slopes indicate that the performance is delayed. Zero slope means that the source performance is on time.
3. Visualization of both MoCaps with heat maps. The timing of each frame is represented by lines following the same color scales that the first visualization. Two consecutive lines with the same slope indicate that the source performance is on time; however, if the slope of consecutive lines differs, the performance is delayed or hastened.

These three representations of the time alignment of two MoCaps permit to interpret how synchronized the two participants are and how consistent one participant is in performing the sequence of movements.

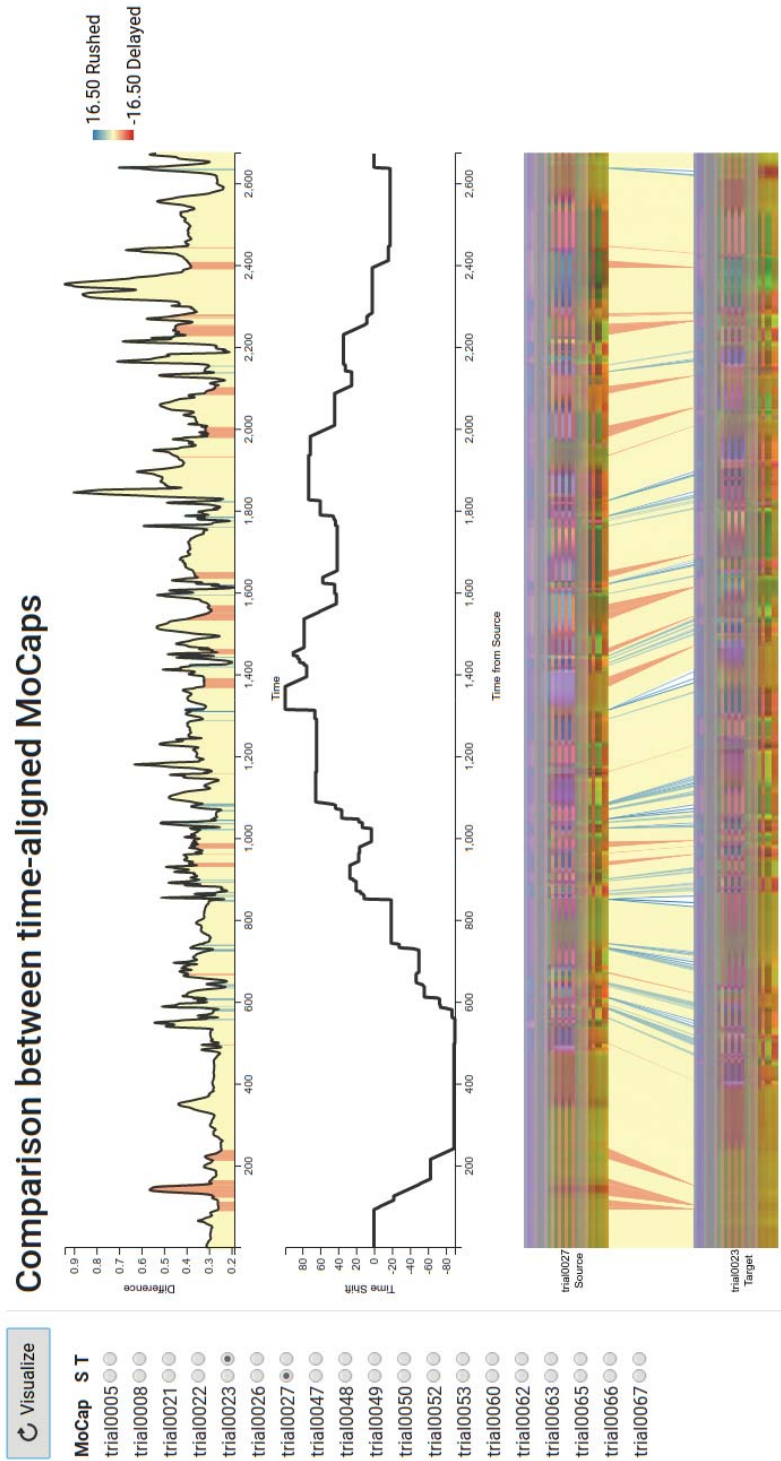


Fig. 4: The left panel shows the list of MoCaps to compare. Results are visualized to the right: At the top, a graph that shows with colors the misalignment between the MoCaps and the absolute difference between time-aligned frames. The middle graph is the misalignment function. The graph at the bottom shows a heat map visualization of the two MoCaps and how they are aligned in time.



## 6 Implementation notes

The MoCap comparison tool presented in this work was written using Javascript and the D3.js library. A prototype of this applications is available at <http://cs.uns.edu.ar/~dku/mocap-synchromparator>. The input data for the application are the preprocessed results of Section 3 *Data preparation* and the DTW for each possible conversion resulting for Equation 1.

## 7 Conclusions and future work

We have presented a comparative visualization of MoCap sequences that focuses on the time dimension. The visualization gives an overview of the misalignment between two sequences and visual information about when one of them is hastened or delayed with respect to the other. The misalignment color-coding allows the user to spot easily the time frames where the sequences differ. Our tool can be easily extended by showing the corresponding video frame or its related skeleton pose upon user request.

As future work, we plan to improve our tool with suitable interactions for exploration (for example, semantic zoom) and adding the possibility to compare more than two sequences at a time. Regarding the heat map, finding a meaningful order for the markers is somehow desirable. Additionally, we will explore further possibilities with the usage of DTW and comparing the MoCaps considering their actual speed.

**Acknowledgments** This work was funded by PGI 24/ZN33, Secretaría General de Ciencia y Tecnología, Universidad Nacional del Sur, Bahía Blanca, Argentina; and by the European Union’s Horizon 2020 research and innovation programme under grant agreement n. 824160 (EnTimeMent).

## References

1. Assa, J., Caspi, Y., Cohen-Or, D.: Action Synopsis: Pose Selection and Illustration. *ACM Trans. Graph.* 24(3), 667–676 (2005)
2. Bernard, J., Wilhelm, N., Krüger, B., May, T., Schreck, T., Kohlhammer, J.: Motionexplorer: Exploratory search in human motion capture data based on hierarchical aggregation. *IEEE Transactions on Visualization and Computer Graphics* 19(12), 2257–2266 (2013)
3. Bernard, J., Vögele, A., Klein, R., Fellner, D.: Approaches and challenges in the visual-interactive comparison of human motion data. In: *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*. vol. 3, pp. 217–224. SciTePress (2017)
4. Bernard, J., Wilhelm, N., Scherer, M., May, T., Schreck, T.: TimeSeriesPaths: Projection-Based Explorative Analysis of Multivariate Time Series Data. In: *Journal of WSCG*, Vol. 20, No. 2. pp. 97–106 (2012)
5. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*. pp. 359–370. AAAIWS’94, AAAI Press (1994)

6. Bruderlin, A., Williams, L.: Motion signal processing. In: Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques. pp. 97–104. SIGGRAPH '95, ACM (1995)
7. Burger, W., Burge, M.J.: Principles of Digital Image Processing. Core Algorithms. Springer-Verlag London (2009)
8. Cho, K., Chen, X.: Classifying and visualizing motion capture sequences using deep neural networks. CoRR abs/1306.3874 (2013)
9. Hu, Y., Wu, S., Xia, S., Fu, J., Chen, W.: Motion track: Visualizing variations of human motion data. In: 2010 IEEE Pacific Visualization Symposium (PacificVis). pp. 153–160 (2010)
10. Jang, S., Elmqvist, N., Ramani, K.: MotionFlow: Visual Abstraction and Aggregation of Sequential Patterns in Human Motion Tracking Data. IEEE Transactions on Visualization and Computer Graphics 22(1), 21–30 (2016)
11. Jiang, J., Xing, Y., Wang, S., Liang, K.: Evaluation of robotic surgery skills using dynamic time warping. Computer Methods and Programs in Biomedicine 152(Supplement C), 71 – 83 (2017)
12. Kolykhalova, K., Camurri, A., Volpe, G., Sanguineti, M., Puppo, E., Niewiadomski, R.: A multimodal dataset for the analysis of movement qualities in karate martial art. In: Proceedings of the 2015 7th International Conference on Intelligent Technologies for Interactive Entertainment (INTETAIN). pp. 74–78. INTETAIN '15, IEEE Computer Society (2015)
13. Krüger, B., Tautges, J., Weber, A., Zinke, A.: Fast local and global similarity searches in large motion capture databases. In: Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. pp. 1–10. SCA '10, Eurographics Association (2010)
14. Li, W., Bartram, L., Pasquier, P.: Techniques and approaches in static visualization of motion capture data. In: Proceedings of the 3rd International Symposium on Movement and Computing. pp. 14:1–14:8. MOCO '16, ACM (2016)
15. Malmstrom, C., Zhang, Y., Pasquier, P., Schiphorst, T., Bartram, L.: MoComp: A Tool for Comparative Visualization Between Takes of Motion Capture Data. In: Proceedings of the 3rd International Symposium on Movement and Computing. pp. 11:1–11:8. MOCO '16, ACM (2016)
16. Müller, M.: Information Retrieval for Music and Motion. Springer-Verlag (2007)
17. Niewiadomski, R., Kolykhalova, K., Piana, S., Alborn, P., Volpe, G., Camurri, A.: Analysis of Movement Quality in Full-Body Physical Activities. ACM Trans. Interact. Intell. Syst. 9(1), 1:1–1:20 (2019)
18. Rabiner, L., Juang, B.H.: Fundamentals of Speech Recognition. Prentice-Hall, Inc. (1993)
19. Sedmidubsky, J., Elias, P., Zezula, P.: Effective and efficient similarity searching in motion capture data. Multimedia Tools and Applications 77(10), 12073–12094 (2018)
20. Tanisaro, P., Heidemann, G.: Dimensionality reduction for visualization of time series and trajectories. In: Felsberg, M., Forssén, P.E., Sintorn, I.M., Unger, J. (eds.) Image Analysis. pp. 246–257. Springer International Publishing (2019)
21. Vaughan, N., Gabrys, B.: Comparing and combining time series trajectories using dynamic time warping. Procedia Comput. Sci. 96(C), 465–474 (2016)
22. Wilhelm, N., Vögele, A., Zsoldos, R., Licka, T., Krüger, B., Bernard, J.: FuryExplorer: Visual-Interactive Exploration of Horse Motion Capture Data. In: Visualization and Data Analysis (VDA 2015) (2015)